DTIC FILE COPY (4)

AD-A197 742

REPORT DOCUMENTATION PAGE

DTIC ELECTE AUG 0 2 1988 S D

CaD

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) LIDS-P-1790	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Lab. for Inf. and Dec. Systems 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research
6c. ADDRESS (City, State, and ZIP Code) Mass. Institute of Technology Room 35-410, LIDS Cambridge, MA. 02139	7b. ADDRESS (City, State, and ZIP Code) 800 North Quincy Street Arlington, VA. 22217
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-84-K-0519
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

Synthesis of Distributed Command and Control for the Outer Air Battle

12. PERSONAL AUTHOR(S)

Stamos K. Andreadakis and Alexander H. Levis

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical	FROM ____ TO ____	July 1988	13

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The objective is to design distributed Command and Control organizations for the outer air battle. The synthesis problem is formulated as follows: Given the decision-making and information processing necessary for the outer air battle, design the C2 organization that is accurate, timely, exhibits a task throughput rate that is higher than the task arrival rate, and whose decisionmakers are not overloaded. A simple model of the process pertinent to the outer air battle has been developed. The model, although an abstraction of the actual naval air operations, retains the fundamental decision-making features.

A new quantitative methodology for the synthesis of C2 organizations is presented. The methodology consists of four phases: (1) Algorithmic generation of data flow structures in the form of Petri Nets that have specified degrees of redundancy and complexity; (2) Transformation of the data flow structures into decision-making organizations by allocating the functions to individual decisionmakers and then into C2 organizations by incorporating the

(over)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Neil Gerr	22b. TELEPHONE (Include Area Code) (202) 696-4321 22c. OFFICE SYMBOL Code 1111 SP

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

*U.S. Government Printing Office: 1985—507-047

#19 ABSTRACT (cont.)

supporting systems; (3) Evaluation of the resulting designs usign three measures of performance- accuracy, response time, and throughput rate - and a measure of effectiveness; and (4) Modification of the candidate designs to increase their measure of effectiveness.

JULY 1988 LIDS - P - 1790

SYNTHESIS OF DISTRIBUTED COMMAND AND CONTROL FOR THE OUTER AIR BATTLE*

Stamos K. Andreadakis
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

The objective is to design distributed Command and Control organizations for the outer air battle. The synthesis problem is formulated as follows: Given the decision-making and information processing necessary for the outer air battle, design the C^2 organization that is accurate, timely, exhibits a task throughput rate that is higher than the task arrival rate, and whose decisionmakers are not overloaded. A simple model of the processes pertinent to the outer air battle has been developed. The model, although an abstraction of the actual naval air operations, retains the fundamental decision-making features.

A new quantitative methodology for the synthesis of C^2 organizations is presented. The methodology consists of four phases: (1) Algorithmic generation of data flow structures in the form of Petri Nets that have specified degrees of redundancy and complexity; (2) Transformation of the data flow structures into decision-making organizations by allocating the functions to individual decisionmakers and then into C^2 organizations by incorporating the supporting systems; (3) Evaluation of the resulting designs using three measures of performance - accuracy, response time, and throughput rate - and a measure of effectiveness; and (4) Modification of the candidate designs to increase their measure of effectiveness.

* This work was conducted at the MIT Laboratory for Information and Decision Systems with support provided by the Office of Naval Research under Contract no. N00014-84-K-0519 (NR 649 003).

SYNTHESIS OF DISTRIBUTED COMMAND AND CONTROL
FOR THE OUTER AIR BATTLE *

Stamos K. Andreadakis
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology, Cambridge, MA.

ABSTRACT

The objective is to design distributed Command and Control organizations for the outer air battle. The synthesis problem is formulated as follows: Given the decision-making and information processing necessary for the outer air battle, design the C^2 organization that is accurate, timely, exhibits a task throughput rate that is higher than the task arrival rate, and whose decisionmakers are not overloaded. A simple model of the processes pertinent to the outer air battle has been developed. The model, although an abstraction of the actual naval air operations, retains the fundamental decision-making features.
A new quantitative methodology for the synthesis of C^2 organizations is presented. The methodology consists of four phases: (1) Algorithmic generation of data flow structures in the form of Petri Nets that have specified degrees of redundancy and complexity; (2) Transformation of the data flow structures into decision-making organizations by allocating the functions to individual decisionmakers and then into C^2 organizations by incorporating the supporting systems; (3) Evaluation of the resulting designs using three measures of performance - accuracy, response time, and throughput rate - and a measure of effectiveness; and (4) Modification of the candidate designs to increase their measure of effectiveness.

1. INTRODUCTION

Organizations are formed when the task to be performed exceeds the capabilities of a single decisionmaker. Even when a single person can complete the task, he may not be able to produce a satisfactory response within the time limits imposed by the task, and keep up with the arrival rate of the tasks. The organization designer is faced with the problem of designing an organization that will meet these design specifications and, in addition, assign subtasks or functions to members of the organization so that no one is overloaded. The design has to be robust to accommodate the decision-making styles of different actual decisionmakers that may instantiate the organization at different times.

Consider, for example, the design of an air-traffic control center for a busy airport area. The task cannot be performed by a single controller, several controller stations may be required. The designer has to take into account the uncertainty that is inherent in the task, the need for accurate and timely responses by the controllers, and the need to keep up with the rate of the incoming tasks. But he also has to consider that different controllers will be on duty at any instant of time. While they are all well trained for their tasks, their actual information processing rate and decision-making styles will differ.

The quantitative and qualitative analysis and evaluation of such task-oriented organizations has been the subject of recent research: Drenick (1986); Levis (1984). In the latter work, a model of the interacting decisionmaker with bounded rationality was introduced by Boettcher and Levis (1982), in which the individual members' cognitive workload was computed using N-dimensional Information theory and the Partition Law of Information (Conant, 1976). The organizational architecture, i.e., the allowable interactions among decisionmakers and the protocols that govern them, is described using Petri Nets (Peterson, 1981; Reisig, 1982).

The synthesis of Command and Control organizations is a complex process that must address a multitude of issues: specifically, how to partition the task into functions (or subtasks), how many decisionmakers to select, how to allocate the functions to decisionmakers, how to select the schema of information exchange among the decisionmakers (protocols), what kind of communications hardware is required for the timely transmission of information and data, what the structure of the required databases and the specifications for the respective hardware shoud be, and how to design decision aids and allocate them to the organization members. Finally, there is the issue of evaluation: how to compute the performance and the effectiveness of the designs and how to select the best design for the task. Consequently, it is necessary to develop a methodology so that the design of C^2 organizations becomes a structured process.

In this paper, a methodology for the synthesis of C^2 organizations is presented. The approach taken in this work decouples the decomposition of the decision-making process and the exchange of data among the functions from the allocation of functions to decisionmakers, and the selection of the supporting systems. Thus, the methodology tackles the synthesis problem at two levels: the *data flow structure level* and the *organization architecture level.*

An algorithm for the generation of data flow structures has been developed; the data flow structures are parameterized by the degree of complexity of the information processing and the degree of redundancy of the information within the structure. The data flow structures are transformed into organization architectures by allocating the functions to decisionmakers and by augmenting the structures to incorporate the supporting systems.

A procedure for the analysis and evaluation of organizational designs is described. The following measures of performance

* This work was supported by the Office of Naval Research under contract No. N00014-K-84-0519 (NR 649-003)

(MOPs) are computed: Accuracy, denoted by the Cost index J; Response Time (Time Delay), denoted by T_r; and Throughput Rate, denoted by R. A global measure, called the measure of effectiveness (MOE), is defined, and the designs are evaluated and compared on the basis of their MOE value.

The processing times associated with the individual information processing and decision-making functions are characterized by probability density functions (pdfs). A method for the computation of the pdf of the organization's response time, and of the pdf of the throughput rate is presented. These lead to the definition of a measure of timeliness and a measure of processing capacity.

2. ORGANIZATION MODEL

2.1 Mathematical Representation

Decision-making organizations can be represented by Petri Nets. In Petri Nets, there are two types of nodes: places, denoted by circles representing signals or conditions; and transitions, denoted by bars, representing processes or events. Places can only be connected to transitions, and transitions can only be connected to places. The execution of a Petri Net is controlled by tokens, which are markers, denoted by dots in the places. A Petri Net is said to execute when a transition fires. A transition can fire only when it is enabled, i.e., when all its input places contain at least one token each. When a transition fires, it removes (consumes) one token from each of its input places and creates (deposits) one token in each of its output places.

A transition may have more than one output places. However, to model decision-making, it is convenient to introduce a special transition*, a decision switch (Figure 1), in which the output places represent alternatives (Tabak and Levis, 1985). When a decision switch fires, a token is deposited in *only one* of its output places. A decision rule associated with this special transition determines the place in which the token is deposited. The rule can be deterministic or stochastic; it can be independent of the attributes of the tokens in the input places or it may depend on them.

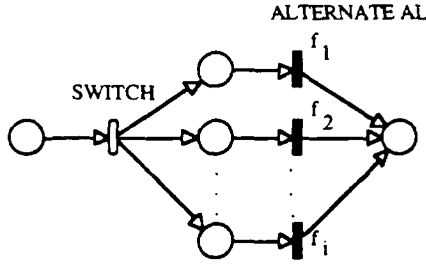

Figure 1. Decision Switch

For example, a decisionmaker may access a decision aid; the decision in this case is whether to access or not the decision aid. Similarly a decisionmaker may have two algorithms or procedures available to process the information. In the case of situation assessment, one algorithm may be complex and the other simple. In the case of developing courses of action, one algorithm may be more detailed and exhaustive, while the other more crude and simplistic. The decision in this case is which of the two algorithms to use. This decision varies among decisionmakers (difference in style) and may be quantified by the relative frequency of algorithm selection. The decisionmaker's workload, the accuracy, the response time, and the throughput rate of the organization are affected by these decisions.

* The use of a special transition can be avoided, if Predicate Transition Nets are used in place of ordinary Petri Nets. However, such a generalization is not necessary for this work.

The decisionmakers are assumed to be limited in the number of functions (or subtasks) they may perform at the same time. This limitation is represented by introducing a place for each decisionmaker, called the resource availability place, which is an output place of the last transition and an input place of the first transition assigned to the decisionmaker. The number of functions a decisionmaker may perform at the same time are represented by the number of tokens initially deposited in the resource availability place, i.e., by its initial marking.

2.2 Decision-making Model and Decision Strategies

The decisionmakers are assumed to be well trained and to be able to use more than one procedure to perform some functions (Boettcher and Levis, 1982; Levis and Boettcher, 1983). The selection of one procedure from a set of procedures is modeled in the Petri Net by a switch. When the decisionmaker selects always the same algorithm (when the selection rule is independent of the input) or the same algorithm for each input element (when the selection rule is dependent on the input value), he is implementing a *pure decision strategy*.

Mixed strategies are obtained when the decisionmaker uses a mix of pure strategies (Owen,1968); the mixed strategies are characterized by the relative frequency of use of the pure strategies. When all decisionmakers of the organization select (implement) their mixed strategies, a *behavioral decision strategy* is obtained.

If the probabilities that correspond to the relative frequency of algorithm use are discretized, then a finite number of mixed strategies is defined for each decisionmaker, and consequently, a finite number of behavioral strategies for the organization is obtained.

3. ANALYSIS OF DECISION-MAKING ORGANIZATIONS

3.1 Workload and Bounded Rationality

Workload represents the amount of mental effort expended by the decisionmakers in order to perform their assigned tasks. The analytical framework for the computation of a surrogate for workload is N-dimensional information theory (Reisbeck, 1963, Shannon and Weaver, 1963). This surrogate, denoted by G, is the total activity term in the Partition Law of Information (Conant, 1976). The total activity has units of bits/symbol.

The value of G depends on several factors. First, it depends on the uncertainty of the organization's task, as modeled by the probability distribution p(x) associated with the input set [x]. It depends also on the structure of the organization - the interactions among decisionmakers - and on the algorithms used to represent the various processing functions, such as situation assessment, courses of action development, and response selection. Finally, it depends on the internal decision strategies of each individual decisionmaker. Indeed, in the analysis of organizational performance that follows, for a given organizational design, the independent variables are the decisions of each decisionmaker and the dependent variables are the workload and the measures of performance.

The qualitative notion that the rationality of the human decisionmaker is bounded, (March, 1978), has been modeled as

$$F \leq F_0 \tag{1}$$

where F is the information processing rate of individual decisionmakers (in bits/sec), and F_0 is the maximum information processing rate that characterizes individual decisionmakers. Since the processing time, t, is computed by

$$t = G/F \tag{2}$$

the minimum processing time t_0, corresponds to the maximum processing rate F_0.

$$t_0 = G/F_0 \tag{3}$$

In evaluating decision-making organizations, it is of interest to compute the minimum response time, and the maximum throughput rate, that correspond to the maximum information processing rate F_0.

The maximum processing rate F_0 varies among decisionmakers. If the pdf $h(F_0)$ of F_0 is known, then the pdf $q(t_0)$ of the minimum processing time of each transition can be obtained:

$$q(t_0) = (G/t_0^2)\, h(G/t_0) \tag{4}$$

3.2 Measures of Performance

The measures of performance considered in this paper are: accuracy, response time, and throughput rate.

Accuracy quantifies the degree to which the actual organization response, Y_j, matches the desired or ideal response Y_{dj}. A cost $C(Y_j,Y_{dj})$ is assigned to the discrepancy of Y_j and Y_{dj} (Levis, 1984). This cost is computed for each input task, x, and each decision strategy. The accuracy measure J is the expected value of the cost and is computed using the probability distribution of the input tasks (Figure 2).

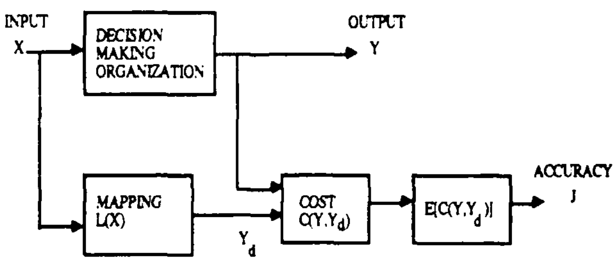

Figure 2. Computation of Accuracy

The response time or time delay of an organization is the time elapsed between sensing the input and producing an output. The expected response time (expected time delay) is a measure of performance that can be used to assess the timeliness of an organization's response.

Timeliness expresses the ability of organizations to produce a response to a given input within an allotted time. The allotted time is a time interval (T_{min}, T_{max}). T_{min} is a time threshold such that if the organization acts in response to the input before the threshold, a cost is incurred through the expense of assets or supplies too early, resulting in the decrease of the probability of success of the response. T_{max} is a time threshold such that if the organization acts in response to the input after the threshold, there will not be enough time left for the implementation of the response. If the expected response time is within the interval (T_{min}, T_{max}) the response is timely. However, this measure of performance does not take into account the variance of the response time. A better measure of timeliness, T, is the probability that the response time, T_r, lies inside the interval (T_{min}, T_{max}), i.e.

$$T = P(T_{min} < T_r < T_{max}) \tag{5}$$

The Throughput rate of the organization is the maximum task processing rate that can be sustained, without queueing of the inputs, or queueing of information at any stage of processing. For the case of stochastic processing times, the pdf of the throughput rate, R, can be computed and a measure of processing capacity, S, can be defined as:

$$S = P(R > R_0) \tag{6}$$

where R_0 is the task arrival rate. Alternatively, we may be interested in computing the response time and the throughput rate that correspond to the minimum value of the rationality threshold, i.e., $(F_0)_{min}$.

To each behavioral strategy, corresponds a set of values of the measures of performance (MOPs), which defines a vector in the MOP space. Thus, the mathematical models of accuracy, response time, and throughput rate, map the decision strategies into the performance space. As the behavioral strategies change, this vector sweeps a locus in the MOP space, the organization locus. The requirements on the MOPs also define a locus in the MOP space; the requirements locus. Organizational architectures can be evaluated by comparing the organization locus to the requirements locus. Different organizational architectures can be compared on the basis of their corresponding loci.

3.3 Computation of the pdf of Response Time

In the Petri Net representation of an organization, the input (source) and the output (sink) nodes are represented by transitions. Information flow paths are the paths emanating from the source transition and arriving at the output transition. The presence of decision switches in the net, with the position of each switch determined by the internal decision strategies, results in some transitions being active during the processing of any task, and in some being inactive. Therefore for each behavioral strategy, corresponding to pure decision strategies of the decisionmakers, some information flow paths are active (transmitting information) while others are inactive. A set of concurrently active paths is called a complete path.

The simple paths and the complete paths may be identified either by an algorithm developed by Jin (Jin et al., 1986) for acyclical structures, or by an algorithm that computes the elementary directed circuits of the net, developed by Martinez and Silva (1980), and improved by Alaiwan and Toudic (1985).

If a pdf is assigned to the processing time of each processing algorithm, to the transmission delay for each communication process, and to the access time for each decision support system (all of which are represented by transitions on the net), then the pdf of the response time of the organization is computed as follows:

For two cascaded functions with corresponding delay pdfs f(t) and g(t), the total delay is the sum of the two delays. Therefore the pdf h(t) of the total delay is given by the convolution of f(t)

and g(t).

$$h(t) = f(t) * g(t) \tag{7}$$

For an information flow path (if assumed that its transitions do not receive data from other paths), the total delay is the sum of the delays of the individual transitions and consequently the pdf of the total delay is obtained by repeated convolutions.

For two functions which are concurrently active, i.e., on parallel paths (Figure 3), the total delay is the maximum of the corresponding delays. The pdf h(t) of the total delay, if the two delays are independent random variables, is obtained as follows:

$$h(t) = f(t)\, G(t) + F(t)\, g(t) \tag{8}$$

where f(t) and g(t) are the pdfs of each delay and F(t) and G(t) are the corresponding cumulative distribution functions.

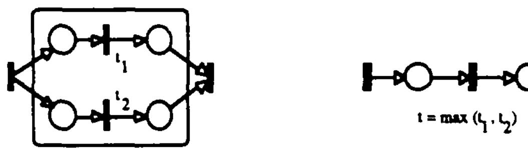

Figure 3. Concurrently Active Functions

For two concurrently active information flow paths, whose transition delays are independent random variables, two cases must be considered:

1) the paths do not have transitions in common
2) the paths have transitions in common

In the first case, the pdf of the total delay of each path is computed, and then the pdf of the maximum of the path delays is obtained. In the second case, if the time delay of the common transition is τ with pdf $f_\tau(t)$, while the total time delays of the unique transitions on each path are τ_1 and τ_2, with corresponding pdfs $g_{\tau 1}(t)$ and $g_{\tau 2}(t)$, first compute the pdf $g_{\tau max}(t)$ of the maximum delay $\tau_{max} = \max(\tau_1, \tau_2)$ of the unique transitions

$$g_{\tau max}(t) = g_{\tau 1}(t)\, G_{\tau 2}(t) + G_{\tau 1}(t)\, g_{\tau 2}(t) \tag{9}$$

and next compute the pdf of the total delay of the two paths by convolving $f_\tau(t)$ and $g_{\tau max}(t)$. Using these procedures, the pdf of each complete path is computed.

Consider two alternate procedures (i.e. substitutes for one another) or two complete paths active with relative frequency of use p_1 and $p_2 = 1-p_1$, and corresponding delay pdfs f(t) and g(t). Then the pdf h(t) of the total delay is given by:

$$h(t) = p_1\, f(t) + p_2\, g(t) \tag{10}$$

3.4 Computation of the pdf of Throughput Rate

The throughput rate of the organization is equal to the minimum of the processing rates of the sets of functions performed by individual decisionmakers, and the processing rates of sets of functions performed by several decisionmakers in an interleaved pattern. These sets of functions correspond to the transitions of the directed elementary circuits of the net. The processing rate of a directed elementary circuit is the inverse of the total processing time of the circuit. The total processing time, t, of a directed elementary circuit is equal to the sum of the processing times of its transitions, divided by the token content, C, of the circuit (Ramchandani, 1974). The token content is equal to the sum of the tokens initially placed in the resource availability places of the circuit.

$$t = (\Sigma t_i) / C \tag{11}$$

In the case of stochastic processing times, the pdf g(r) of the processing rate r of a directed elementary circuit is given by

$$g(r) = (1/r^2)\, f(1/r) \tag{12}$$

where f(t) is the pdf of the processing time of the circuit. For two directed elementary circuits with no transitions in common, if the processing rates r_1 and r_2 are independent random variables with pdfs f(r) and g(r), the pdf of the minimum processing rate h(r) is

$$h(r) = f(r)\, [1-G(r)] + [1-F(r)]\, g(r) \tag{13}$$

where F(r) and G(r) are the cumulative distribution functions.

If the elementary circuits have transitions in common, then their processing rates are correlated. Let two such circuits have one transition in common with processing time τ, having pdf g(t), and one unique transition in each circuit with corresponding processing times τ_1 and τ_2, with pdfs $q_{\tau 1}(t)$ and $q_{\tau 2}(t)$ (Figure 4). Assume also that the two circuits have the same token content C. Then the pdf of the maximum processing times of the two circuits

$$\tau_{max} = (\tau + \max(\tau_1, \tau_2)) / C \tag{14}$$

is computed as follows: first compute the pdf $q_{\tau *}(t)$ of $\tau^* = \max(\tau_1, \tau_2)$ by:

$$q_{\tau *}(t) = q_{\tau 1}(t)\, Q_{\tau 2}(t) + Q_{\tau 1}(t)\, q_{\tau 2}(t) \tag{15}$$

Then convolve $q_{\tau *}(t)$ and g(t)

$$s(t) = q_{\tau *}(t) * g(t) \tag{16}$$

and scale the pdf s(t) to obtain the pdf $f_{tmax}(t)$ of the maximum processing time of the two circuits

$$f_{tmax}(t) = C\, s(Ct) \tag{17}$$

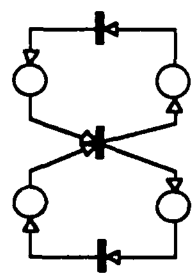

Figure 4. Two circuits with one common transition

Finally, the pdf h(r) of the minimum of the processing rates of the two circuits is

$$h(r) = (1/r^2)\, f_{tmax}(1/r) \tag{18}$$

The computation in the case of different token content is similarly developed.

3.5 Measure of Effectiveness

Measures of Effectiveness quantify the degree to which an organization meets its requirements. A Measure of Effectiveness, Q, can be defined by the ratio of the number of behavioral strategies that satisfy the requirements to the total number of behavioral strategies.

$$Q = \frac{\text{number of behavioral strategies satisfying the requirements}}{\text{total number of behavioral strategies}} \tag{19}$$

Recall that individual decisionmakers differ in style, i.e., they tend to use different mixed strategies. In this respect the measure of effectiveness is a measure of robustness of the organization with respect to the different styles of individual decisionmakers.

4. SYNTHESIS OF DECISION-MAKING ORGANIZATIONS

Given a complex information processing and decisionmaking task, there exists a multitude of ways to partition the processing of a task into subtasks (functions), to define the schema of information exchange among the functions, to allocate functions to decisionmakers, and to specify the supporting systems (software and hardware).

4.1 Synthesis Problem Formulation

The synthesis problem is formulated as follows: Given a mission and a set of tasks to be performed, design a decision-making organization that is accurate, timely, has a task throughput rate higher than the task arrival rate, and whose decisionmakers are not overloaded (Andreadakis, 1988). The quantitative formulation is:

Accuracy greater than or equal to a given threshold, or equivalently, expected cost J less than or equal to some threshold J_o:

$$J \leq J_o \tag{20}$$

Timeliness measure greater than or equal to some threshold T_o:

$$T \geq T_o \tag{21}$$

Processing capacity measure greater than or equal to some threshold S_o:

$$S \geq S_o \tag{22}$$

under the constraint that decisionmakers are not overloaded, i.e., that each decisionmaker's information processing rate is less than or equal to his rationality threshold $(F_o)_i$:

$$F_i \leq (F_o)_i \tag{23}$$

An alternative formulation is obtained when the second and third requirements are expressed as:

Response time T_r less than or equal to some threshold $(T_r)_o$:

$$T_r \leq (T_r)_o \tag{24}$$

Throughput rate R greater than the task arrival rate R_o:

$$R > R_o \tag{25}$$

In this work the concepts of *data flow structure* (DFS), *decision-making organization* (DMO) and *Command and Control organization* (C^2O) are contrasted, and are employed in the development of a structured methodology for the synthesis of Command and Control organizations.

The DFS is a representation of the connectivity of the functions performed by the organization and illustrates the flow of information from function to function. The DMO is a DFS whose functions have been allocated to decisionmakers. Finally, a C^2O is a DMO which is supported by hardware and software (the C^3 system) in the execution of its tasks.

In the two level design procedure, the data flow structure design focuses on information processing schemata, while the organization architecture design focuses on function allocation to decisionmakers and on the development of the supporting systems.

The synthesis methodology has four phases (Figure 5). In phase 1, the procedure for generating data flow structures produces a set of candidate designs. In phase 2, each data flow structure is augmented and transformed to one or more decision-making organizations, in which the functions have been allocated to decisionmakers, and then to the corresponding

Command and Control organizations by incorporating the supporting hardware and software. In phase 3, the measures of performance and the measure of effectiveness are computed. The designs obtained in this manner, are revised in phase 4, to increase their measure of effectiveness by changing function allocation, introducing or modifying decision aids and

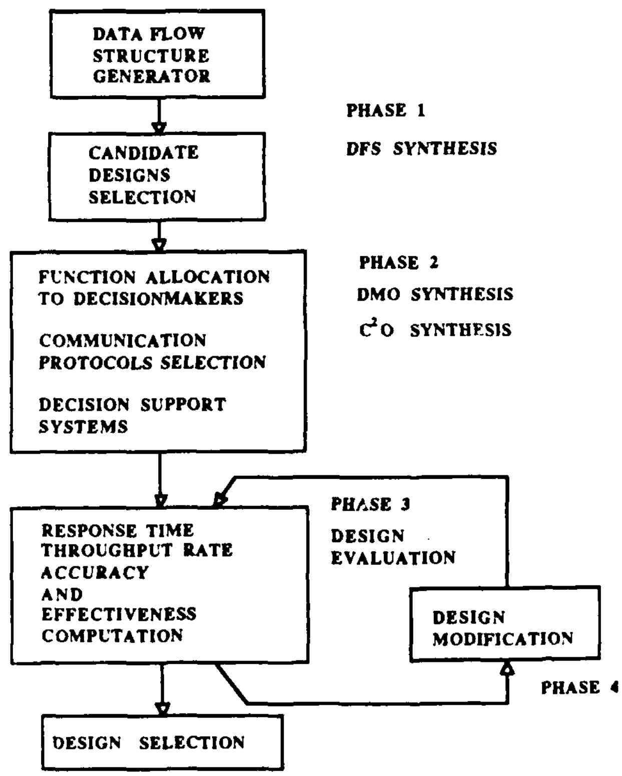

Figure 5. Flowchart of Synthesis Methodology

databases, and improving the communication links. Finally, a C^2 organization is selected on the basis of the highest MOE value.

4.2 Data Flow Structure Design

The information processing is decomposed into five stages (functions): Initial Processing [IP], Data Fusion [DF], Middle Processing [MP], Results Fusion [RF], and Final Processing [FP]. As data are received, they are processed in the IP stage to assess the situation. Information (local or partial situation assessments) of several IP stages are combined (fused) in the DF stage, which produces global situation assessment.

The global situation assessment is fed to the MP stage which develops results (options or courses of action). The results are combined (fused) in the RF stage to eliminate conflicting or infeasible options - courses of action. Finally, a response is selected from the available options in the FP stage.

Each processing stage is represented in the Petri Net of the data flow structure by a transition. An information flow path with all five stages defines a flow type 1 (Figure 6a). Note that some IP transitions may provide results for fusion at an RF stage (DF and MP stages null) (Figure 6c), while some MP transitions may generate output of the organization (RF and FP stages null) (Figure 6b). An information flow path of the latter type defines flow type 2, while one of the former type defines a flow type 3.

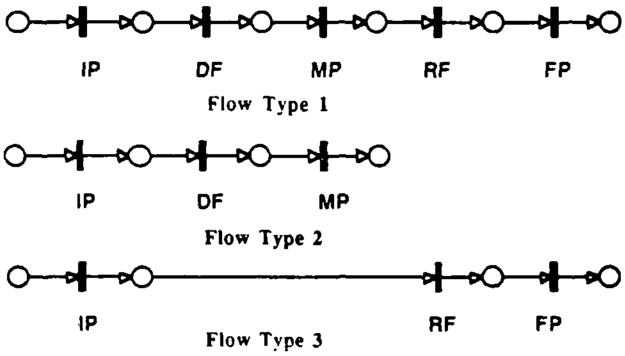

Figure 6. Basic Flow Types

The data flow structures are classified according to the flow types of their information flow paths. If all the paths are of flow type 1, then the DFS belongs to class 1. If some paths are of flow type 1 and some of flow type 2, the DFS class is 12. The feasible classes are: 1, 2, 3, 12, 13, and 123. Class 23 is infeasible because the flow type 2 information paths have data for fusion and DF transitions, while the flow type 3 information paths have results for fusion and RF transitions; and hence flow type 2 and flow type 3 paths cannot exchange information. A DFS with all three flow types (class 123) is shown in Figure 7.

The grammar rules for the connectivity of the processing transitions are:

- exactly one MP node can receive data from a DF node
- exactly one FP node can receive data from an RF node
- one IP transition for each input to the organization
- one FP transition for each output of the organization

The generation of data flow structures takes into account the complexity and redundancy of information processing that is required by the task, and the organization's objectives.

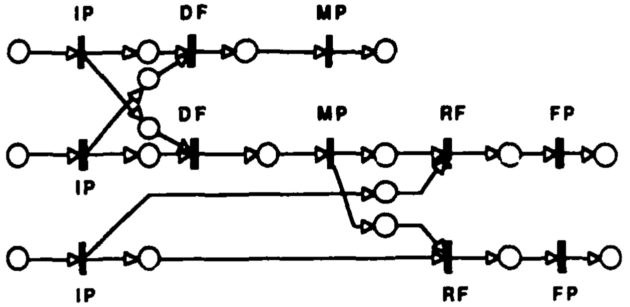

Figure 7. Data Flow Structure with all three Flow Types.

Depending on the degree of centralization of decision-making for global situation assessment and the magnitude of the geographical area for which global situation assessment is desired, the data fusion stage may be more or less complex. Similarly, depending on the degree of centralization for global response selection, and the magnitude of the geographical area where the response needs to be coordinated, the results fusion stage may be more or less complex.

The *degree of complexity* of a DF transition is defined as the number of transitions that feed data to the DF transition. The degree of complexity of the DF stage is defined as the maximum of the degrees of complexity of the DF transitions. The term complexity is justified by the observation that the more data that are fed to a fusion node, the more complex the processing that takes place.

The need for redundancy of information within the structure arises from survivability considerations and topological factors. The *degree of redundancy* of an IP transition is defined as the number of fusion stages that receive the output data of the IP transition. The degree of redundancy of the DF stage is defined as the maximum of the degrees of redundancy of the IP transitions. The term redundancy is justified by the fact that the same information is communicated to more than one fusion nodes, and is therefore redundant in the data flow structure.

The degree of complexity of a RF transition, the degree of redundancy of a MP transition, and the degrees of complexity and redundancy of the RF stage are similarly defined. A data flow structure with degree of complexity $c_1 = 2$ and degree of redundancy $r_1 = 2$ of the DF stage, and degree of complexity $c_2 = 3$ and degree of redundancy $r_2 = 3$ of the RF stage is shown in Figure 8.

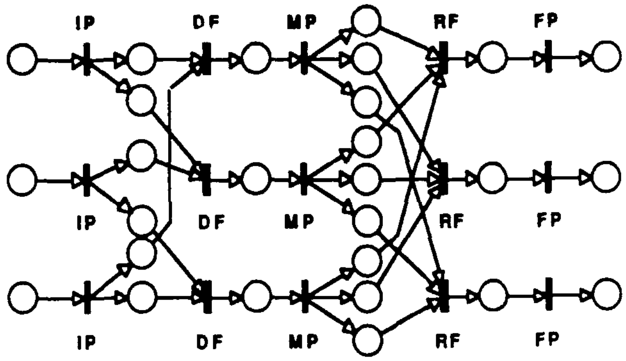

Figure 8. Data Flow Structure $c_1 = 2, r_1 = 2, c_2 = 3, r_2 = 3$

4.3 Data Flow Structure Generation Algorithm

The algorithm for the generation of data flow structures produces the incidence matrix of the corresponding Petri Net, and has seven steps.

The design parameters are:

- the number n_1 of IP transitions that provide data to the DF stage (n_1 less than or equal to the number of IP transitions)
- the number k_2 of MP transitions that provide results to the RF stage (k_2 less than or equal to number of MP transitions)
- the degrees of complexity c_1 and c_2 of the DF and RF stage (less than or equal to the number of transitions that provide information for fusion at the corresponding stage) and
- the degrees of redundancy r_1 and r_2 of the DF and RF stage (less than or equal to the number of processing assets)

Table 1. Transition sets

set	transition type
T_1	initial processing
T_2	data fusion
T_3	middle processing
T_4	results fusion
T_5	final processing

Table 2. Place sets

set	place type
P_1	input places to IP transitions
P_2	output places of IP transitions which are input places to DF transitions
P_3	output places of DF transitions which are input places to MP transitions
P_4	output places of IP transitions which are input places to RF transitions and output places of MP transitions which are input places to RF transitions
P_5	output places of RF transitions which are input places to FP transitions
P_6	output places of MP transitions which are outputs of the DFS
P_7	output places of FP transitions

The incidence matrix has block form: five sets of transitions (Table 1) and seven sets of places (Table 2) are defined. Thus, the incidence matrix is composed of 35 blocks (Figure 9). Each block is denoted by P_iT_j, corresponding to place set P_i and transition set T_j. The flowchart of the algorithm is depicted in Figure 10.

In order to generate data flow structures in a consistent, methodical way, the design parameters are varied between the minimum and maximum value they may obtain.

Step 1: Select the class of the data flow structure.

Step 2: Select the number n_1 of initial processing (IP) transitions that provide data for fusion (DF stage). Let n_2 be the number of initial processing (IP) transitions that provide results for fusion (RF stage). The total number n of IP transitions is:

$$n = n_1 + n_2 \qquad (26)$$

$$
\begin{matrix}
\begin{bmatrix} P_1T_1 \\ n\times n \end{bmatrix} & \begin{bmatrix} P_1T_2 \\ n\times k \end{bmatrix} & \begin{bmatrix} P_1T_3 \\ n\times k \end{bmatrix} & \begin{bmatrix} P_1T_4 \\ n\times m \end{bmatrix} & \begin{bmatrix} P_1T_5 \\ n\times m \end{bmatrix} \\
\begin{bmatrix} P_2T_1 \\ p\times n \end{bmatrix} & \begin{bmatrix} P_2T_2 \\ p\times k \end{bmatrix} & \begin{bmatrix} P_2T_3 \\ p\times k \end{bmatrix} & \begin{bmatrix} P_2T_4 \\ p\times m \end{bmatrix} & \begin{bmatrix} P_2T_5 \\ p\times m \end{bmatrix} \\
\begin{bmatrix} P_3T_1 \\ k\times n \end{bmatrix} & \begin{bmatrix} P_3T_2 \\ k\times k \end{bmatrix} & \begin{bmatrix} P_3T_3 \\ k\times k \end{bmatrix} & \begin{bmatrix} P_3T_4 \\ k\times m \end{bmatrix} & \begin{bmatrix} P_3T_5 \\ k\times m \end{bmatrix} \\
\begin{bmatrix} P_4T_1 \\ q\times n \end{bmatrix} & \begin{bmatrix} P_4T_2 \\ q\times k \end{bmatrix} & \begin{bmatrix} P_4T_3 \\ q\times k \end{bmatrix} & \begin{bmatrix} P_4T_4 \\ q\times m \end{bmatrix} & \begin{bmatrix} P_4T_5 \\ q\times m \end{bmatrix} \\
\begin{bmatrix} P_5T_1 \\ m\times n \end{bmatrix} & \begin{bmatrix} P_5T_2 \\ m\times k \end{bmatrix} & \begin{bmatrix} P_5T_3 \\ m\times k \end{bmatrix} & \begin{bmatrix} P_5T_4 \\ m\times m \end{bmatrix} & \begin{bmatrix} P_5T_5 \\ m\times m \end{bmatrix} \\
\begin{bmatrix} P_6T_1 \\ k_1\times n \end{bmatrix} & \begin{bmatrix} P_6T_2 \\ k_1\times k \end{bmatrix} & \begin{bmatrix} P_6T_3 \\ k_1\times k \end{bmatrix} & \begin{bmatrix} P_6T_4 \\ k_1\times m \end{bmatrix} & \begin{bmatrix} P_6T_5 \\ k_1\times m \end{bmatrix} \\
\begin{bmatrix} P_7T_1 \\ m\times n \end{bmatrix} & \begin{bmatrix} P_7T_2 \\ m\times k \end{bmatrix} & \begin{bmatrix} P_7T_3 \\ m\times k \end{bmatrix} & \begin{bmatrix} P_7T_4 \\ m\times m \end{bmatrix} & \begin{bmatrix} P_7T_5 \\ m\times m \end{bmatrix}
\end{matrix}
$$

Figure 9. Block Form of Incidence Matrix

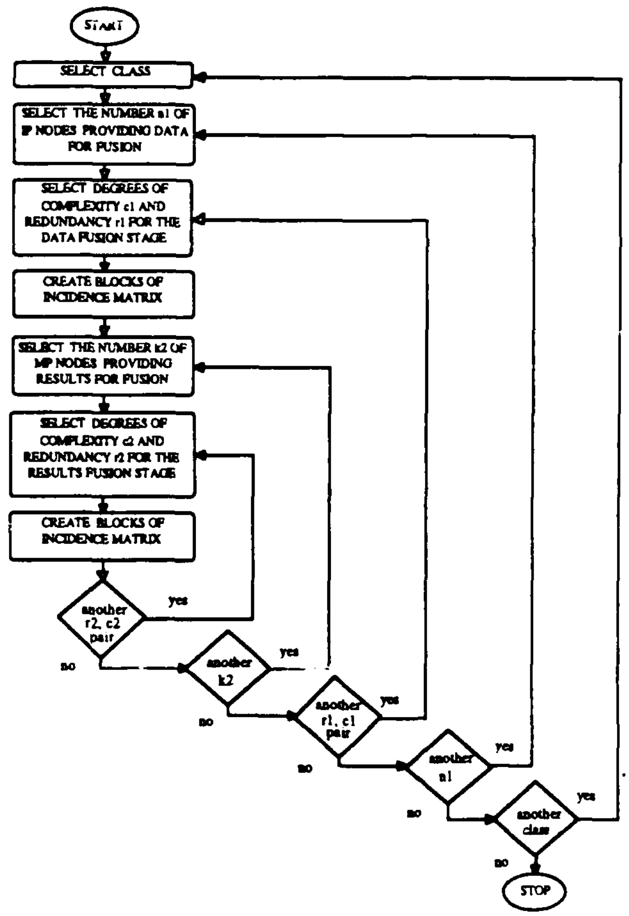

Figure 10. Flowchart of Data Flow Structure Generation Algorithm

Step 3: Select the degree of complexity c_1 and the degree of redundancy r_1 of the DF stage. The number p of output places of IP transitions that belong to the set P_2 is:

$$p = n_1 r_1 \tag{27}$$

and the number k of data fusion transitions is:

$$k = n_1 (r_1 / c_1) \tag{28}$$

For the pair (r_1,c_1) to be feasible, i.e., for all transitions of the stage to have the same degree of complexity and degree of redundancy, the number k must be integer. Another constraint on k is that it be no larger than the number of available processing assets. Since each DF transition is connected to one middle processing (MP) transition, the number of MP transitions is also k.

Step 4: Since one IP transition is connected to each place that represents an input to the organization, and exactly one MP transition is connected to each output place of a DF transition, the diagonal elements of blocks P_1T_1, P_3T_3 are equal to -1, while the non diagonal elements are equal to 0.

Each DF transition has exactly one output place; thus, the diagonal elements of block P_3T_2 are equal to 1, while the non diagonal elements are equal to 0.

The elements B_{ij} of block P_2T_1 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = 1, 2, \ldots, n_1 \text{ and} \\ & i = (j-1)r_1 + s,\ s = 1, 2, \ldots, r_1 \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

The elements B_{ij} of block P_2T_2 obtain their values according to

$$B_{ij} = \begin{cases} -1 & \text{if place i is connected to transition j} \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

The elements of all the other blocks involving place sets P_1, P_2, P_3 are equal to 0.

Step 5: Select the number k_2 of MP transitions that provide results for fusion (at the RF stage). Let k_1 be the number of middle processing transitions that produce outputs. The total number of MP transitions is:

$$k = k_1 + k_2 \tag{31}$$

Step 6: Select the degree of complexity c_2 and the degree of redundancy r_2 of the RF stage. The number q of output places of IP transitions and MP transitions that belong to the set P_4 is:

$$q = (n_2 + k_2)\, r_2 \tag{32}$$

and the number of results fusion transitions, m, is:

$$m = (n_2 + k_2) (r_2 / c_2) \tag{33}$$

For the pair (r_2,c_2) to be feasible, i.e., for all transitions of the stage to have the same degree of complexity and degree of redundancy, m must be integer. The second constraint on m is

$$m \leq a \tag{34}$$

where a is the number of available processing assets. Since each RF transition is connected to one FP transition, the number of FP transitions is also m.

Step 7: The elements B_{ij} of block P_4T_1 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = n_1+1,\ n_1+2, \ldots, n \text{ and} \\ & i = [(j-n_1)-1]r_2 + s,\ s = 1, 2, \ldots, r_2 \\ 0 & \text{otherwise} \end{cases} \tag{35}$$

The elements B_{ij} of block P_4T_3 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = k_1+1,\ k_1+2, \ldots, k \text{ and} \\ & i = [n_2 + (j-k_1)-1]r_2 + s,\ s = 1, 2, \ldots, r_2 \\ 0 & \text{otherwise} \end{cases} \tag{36}$$

The elements B_{ij} of block P_4T_4 obtain their values according to

$$B_{ij} = \begin{cases} -1 & \text{if place i is connected to transition j} \\ 0 & \text{otherwise} \end{cases} \tag{37}$$

Each RF transition has exactly one output place; thus, the diagonal elements of blocks P_5T_4, are equal to 1, while the non diagonal elements are equal to 0.

Each FP transition has exactly one input place; consequently, the diagonal elements of block P_5T_5 are equal to -1, while the non diagonal elements are equal to 0.

Exactly one place representing an output of the DFS is connected to an MP transition which produces a DFS output; likewise, exactly one output place is connected to each FP transition. Hence, the elements B_{ij} of blocks P_6T_3 and P_7T_5, with i = j, are equal to 1, and the other elements are equal to 0.

The elements of all the other blocks involving place sets P_4, P_5, P_6 and P_7 are equal to 0.

4.4 Data Flow Structure Selection

Several data flow structures are generated by the algorithm. In order to select the feasible structures, i.e., those that are appropriate for the task, the designer must consider the suitability of the structure to the information processing required by the task. Consequently, the algorithms that implement the processing functions must be developed, and then be associated with the transitions of each candidate structure. During this stage, some links may be removed from the structure. If it is not possible to associate the algorithms with the transitions of a structure, then the structure is discarded.

4.5 Organization Architecture Design

From each data flow structure, one or more decision-making organizations (DMOs) may be developed through function allocation to decisionmakers. Functions allocated to a decision-maker must observe three requirements:

1) must be connected through an input-output relationship, i.e., the output of the one must be the input to the other, so that the decisionmaker processes information relevant to the same subtask;

2) must belong to different slices (Fernandez and Thiagarajan, 1984) of the Petri Net, so that they observe concurrency; and

3) must conform to the specialization of the decisionmaker.

When a set of functions is allocated to a decisionmaker, a resource availability place is introduced. The addition of these places and of their links, creates the directed elementary circuits of the net, which are used in the throughput rate computations.

The transitions of the DFS are in general macro-transitions; they may have internal structure as in the case of functions performed by alternate algorithms. At this point the macro-transitions are substituted by the subnets that they represent.

Next, each DMO is transformed into a C^2O by incorporating the supporting decision systems and the communication links. The data flow structure is augmented by adding the transitions that represent the communication processes and the decision support systems access, and of the places that represent the corresponding protocols. In general, the decisionmakers may or may not use the decision support systems; therefore switches must be introduced to depict the choices available. The switches and the corresponding strategies enable the modeling of the decision-making styles of individual decisionmakers.

4.6 Design modification rules

If the computed Measure of Effectiveness is not satisfactory, then the organization is modified in order to increase the MOE value. The procedure for the modification depends on the location of the organization locus with respect to the requirements locus. The existing cases are shown in Table 3.

Table 3. Design Modification Cases

case	J<Jo	$T>T_o$ $T_r<T_{ro}$	$S>S_o$ $R>R_o$	must improve	modification required
1	false	true		accuracy	introduce decision aid
2	true	false		response time	better communications improve database access improve decision aids
3	false	false		accuracy and response time	introduce decision aid better communications improve database access
4	true	true	false	throughput rate	modify function allocation more processing channels

5. APPLICATION AND RESULTS

The application of the synthesis methodology will be illustrated through the design of Command and Control organizations for the outer air battle. Three C^2 assets are considered: two airborne warning radar aircraft (E2C) and the Combat Information Center (CIC) on the carrier.

A simple model has been developed for the information processing and decision-making pertinent to the outer air battle. It should be noted that the model is an abstraction of the actual processes and does not necessarily reflect real naval air operations; it can however be modified to represent reality.

The model presumes that the carrier has four squadrons of interceptor aircraft. Two E2Cs are airborne patroling their assigned sectors. One squadron of interceptors is assigned to each E2C, and the other two squadrons are free assets that will be allocated to the appropriate sector(s) depending on the strength of the incoming raid.

The objective of the organization is to develop and implement appropriate plans to engage the incoming threats before they reach the weapons release line. Each of the two E2Cs collects information from the area that it surveils, performs situation assessment, develops courses of action, and selects one response from the developed courses of action. Global considerations necessitate the exchange of information between the two E2Cs and possibly the CIC, in order to resolve conflicting courses of action, to allocate assets, and coordinate the response execution. In this example, the vectoring of interceptors to the threats has not been modeled.

The model used for this example, incorporates the following functions:

- *local (sector) situation assessment*: classification of enemy aircraft based on their signature and air speed, estimation of number of threats and distance from the E2C.
- *global situation assessment:* estimation of raid strength in both sectors.
- *local (sector) courses of action development*: generation of plans -options- depending on the number and type of aircraft in the sector.
- *global response selection (global resources allocation)*: the free assets are assigned to the sectors, or they remain in the inner battle region
- *local (sector) response selection*: one option is chosen from the developed cources of action, given the available assets.

The complete set of data flow structures generated by the algorithm is given in Table 4. Four representative structures are depicted in Figures 11 through 14. Two of these structures, DFS 7 shown in Figure 11, and DFS 11 depicted in Figure 12, will be used to apply phases two and three of the synthesis methodology.

Table 4. Generated Data Flow Structures

class	c_1	r_1	c_2	r_2	DFS
2	2	1			DFS1
2	2	2			DFS2
2	2	3			DFS3
1	2	1	1	2	DFS4
1	2	1	1	3	DFS5
1	2	2	2	1	DFS6
1	2	2	2	2	DFS7
1	2	2	2	3	DFS8
1	2	3	2	2	DFS9
1	2	3	3	1	DFS10
1	2	3	3	2	DFS11
1	2	3	3	3	DFS12
12	2	3	2	1	DFS13
12	2	3	2	2	DFS14

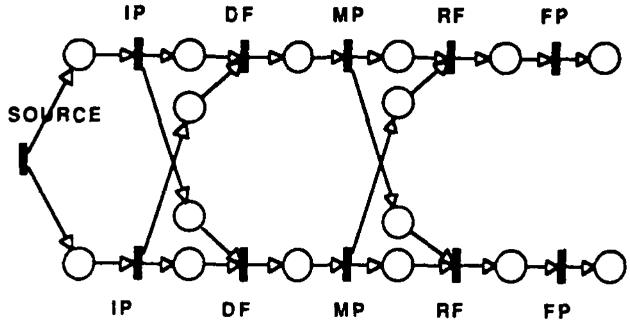

Figure 11. Data Flow Structure 7; $r_1=2, c_1=2, r_2=2, c_2=2$

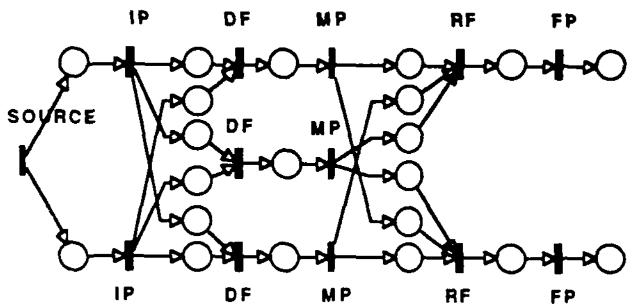

Figure 12. Data Flow Structure 11; $r_1=3, c_1=2, r_2=2, c_2=3$

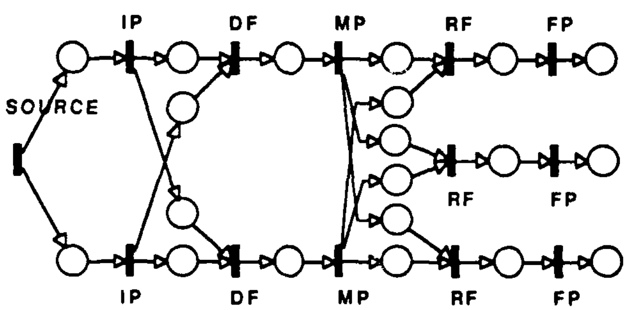

Figure 13. Data Flow Structure 8; $r_1=2, c_1=2, r_2=3, c_2=2$

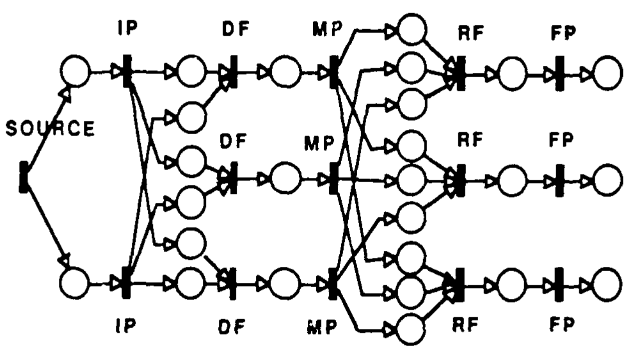

Figure 14. Data Flow Structure 12; $r_1=3, c_1=2, r_2=3, c_2=3$

Two algorithms -procedures- were created for the development of local courses of action: one exhaustive and one crude. Note that the differences in decision-making style, in this example, are manifested in the courses of action development function. The detailed data flow structures, corresponding to Figures 11 and 12, after the elimination of some links that are not required, are shown in Figures 15 and 16.

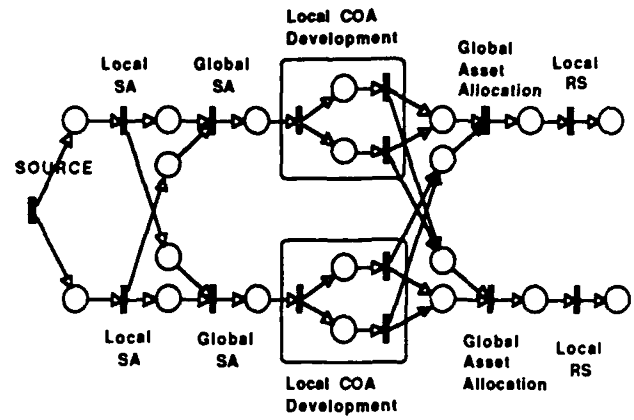

Figure 15. Detailed Data Flow Structure 7

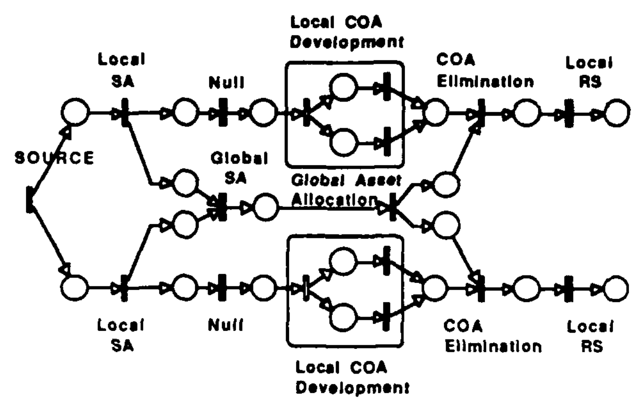

Figure 16. Detailed Data Flow Structure 11

In the data flow structure depicted in Figure 15, the information processing is performed by the personnel of the two E2Cs, while in the data flow structure shown in Figure 16, the CIC personnel participates in the decision-making process by performing the global functions.

From each data flow structure, two Command and Control organizations were developed through different allocation of the functions performed by the E2C personnel: in the first organization, one decisionmaker performs all the functions, while in the second the functions are allocated to two decision-makers in series. The corresponding Command and Control organizations are depicted in Figures 17, 18, 19, and 20.

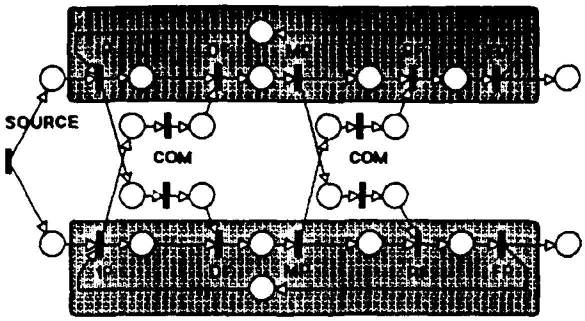

Figure 17. Organization 1; derived from DFS7, one decisionmaker per E2C

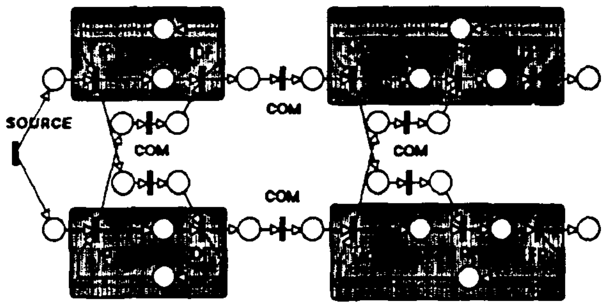

Figure 18. Organization 2; derived from DFS7, two decisionmakeers per E2C

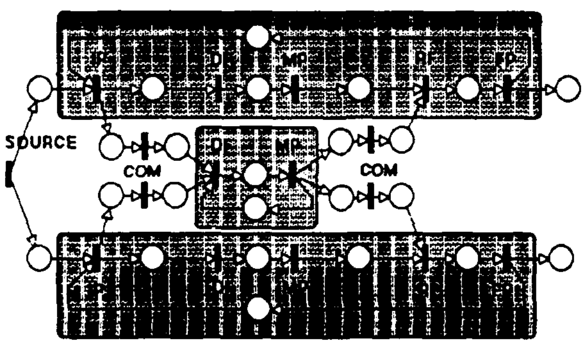

Figure 19. Organization 3; derived from DFS11, one decisionmaker per E2C

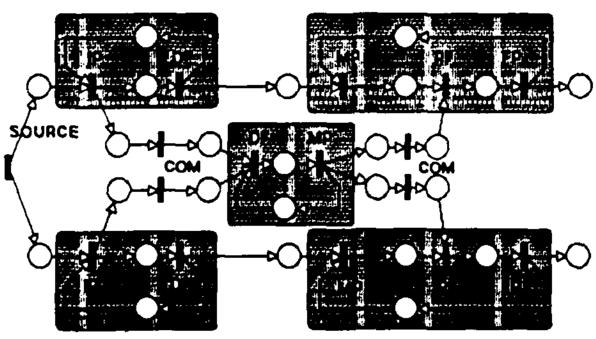

Figure 20. Organization 4; derived from DFS11, two decisionmakers per E2C

The measures of performance, namely the accuracy J, the response time T_r, and the throughput rate R that correspond to the minimum value of the rationality threshold $(F_o)_{min}$, were computed. The value used for $(F_o)_{min}$ is 5 bits/sec (Miller, 1956). Eleven mixed decision strategies were implemented for each MP transition (selecting one of the two COA development algorithms).

$$p_1 = 0.1\,k \qquad k = 0, 1, 2, \ldots, 10 \tag{38}$$

Consequently, the number of behavioral strategies is 121. To each behavioral strategy corresponds a set of MOP values. The ranges of the MOPs are shown in Figures 21, 22, and 23.

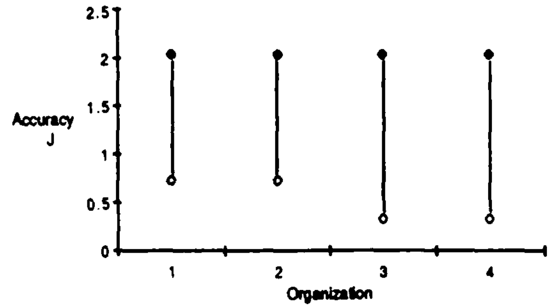

Figure 21. Range of Accuracy

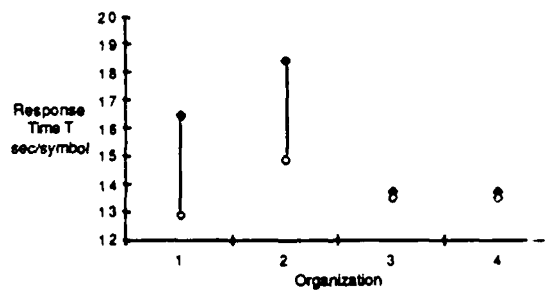

Figure 22. Range of Response Time

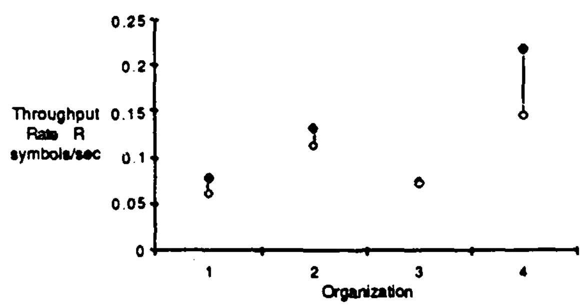

Figure 23. Range of Throughput Rate

Organization 2 (4 DMs) has greater response time than organization 1 (2 DMs) due to the time-consuming communication between the decisionmakers of each E2C. It has also greater throughput rate due to the function allocation to two decisionmakers, which creates directed elementary circuits with smaller processing times.

Organizations 3 and 4 have the same response time for corresponding behavioral strategies, because the information flow path that is dominant (critical circuit) does not contain the communication process between the two decisionmakers of the E2C. Organization 4 (5 DMs) has greater throughput rate than organization 3 (3 DMs), due to the allocation of functions to two decisionmakers per E2C.

The measure of Effectiveness Q has been parameterized by the requirements on accuracy, J_0, response time, T_0, and throughput rate, R_0. The computed accuracy of all organizations is comparable. Thus, a qualitative comparison can be performed by studying the sensitivity of Q to the requirements on response time, T_0, and throughput rate, R_0 (Figures 24, 25, 26, and 27). These plots show the value of the Measure of Effectiveness for all combinations of requirements on the response time and the throughput rate, when the requirement on the accuracy is fixed at $J_0 = 2.04$.

As the requirement on response time becomes more stringent, organization 2 has fewer behavioral strategies that satisfy the requirement than organization 1, i.e., lower MOE value. Conversely, when the requirement on throughput rate is increased, Organization 2 has more behavioral strategies satisfying the requirement than organization 1, and thus higher MOE value. When the requirement on throughput rate increases, Organization 4 has higher MOE value than organization 3. Finally, it is observed that organization 4 has a higher MOE value for all sets of requirement values than the other organizations; thus organization 4 is the best overall design.

The synthesis methodology, when applied to a problem such as this one in which it is necessary to process large amounts of information and arrive at accurate decisions in a timely manner, has yielded a number of data flow structures. From these structures, alternative organization architectures were obtained. The evaluation of the organizations in terms of their effectiveness has led to the desired result - a preferred organizational design.

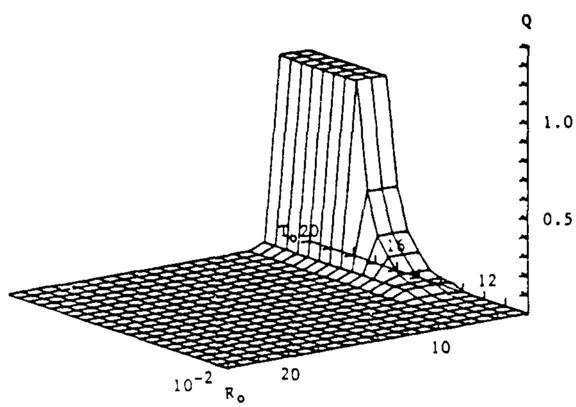

Figure 24. MOE Q of organization 1 vs requirements T_0 and R_0

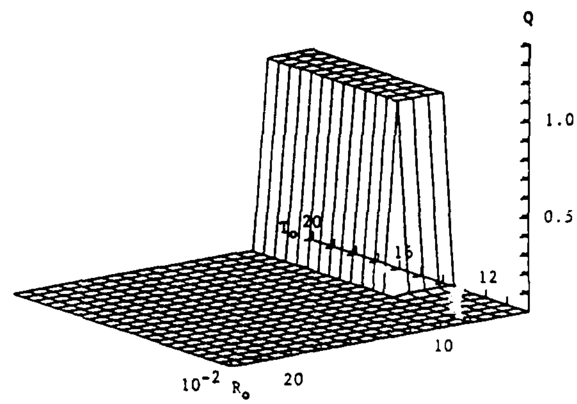

Figure 26. MOE Q of organization 3 vs requirements T_0 and R_0

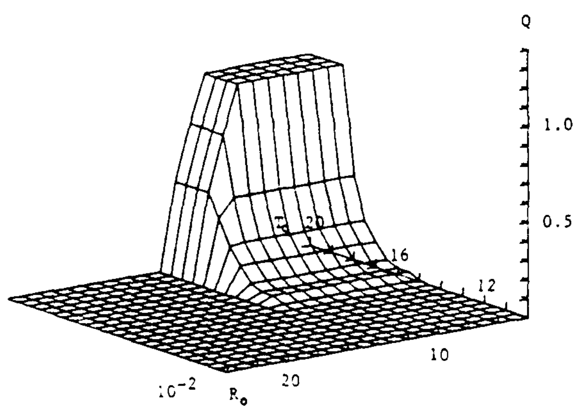

Figure 25. MOE Q of organization 2 vs requirements T_0 and R_0

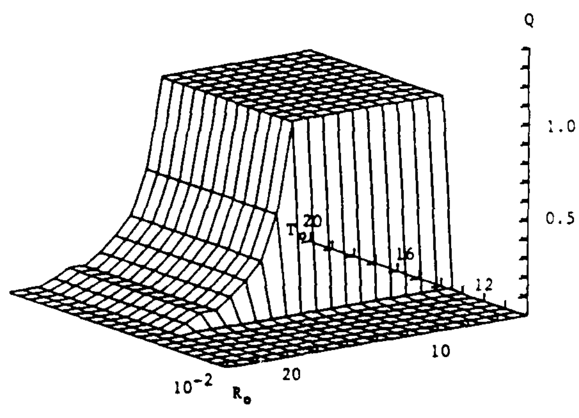

Figure 27. MOE Q of organization 4 vs requirements T_0 and R_0

6. CONCLUSIONS

Four Command and Control organizations have been developed for the outer air battle, using a structured synthesis methodology. The synthesis methodology tackles the design problem at two levels: the data flow structure level and the organization architecture level. This decoupling enables the formulation of an algorithm that generates data flow structures parameterized by the complexity and redundancy of the information processing, without consideration of the organizational constraints. The organization architectures are developed from the candidate data flow structures, through the allocation of functions to decisionmakers, and the selection of the supporting software and hardware.

The quantitative analysis computes the MOPs and MOE , taking into consideration both the variance of the human decisionmakers' maximum information processing rate, and differences in decision-making style. The organizations are compared on the basis of their MOE, which is a measure of robustness of the design to the strategies implemented by individual decisionmakers instantiating the organization.

The qualitative analysis of the sensitivity of the MOE to the requirements on the MOPs, allows for the selection of the best design, i.e. the organization that satisfies the requirements and is more robust to the decision-making styles of the organization members.

The methodology is a flexible top-down approach to the design problem, that results in the expansion of the set of candidate architectures. A potential benefit from the top-down approach is that the requirements for decision aids, databases, and communication links may be derived through the objective evaluation of the effectiveness of the C^2 organization.

Finally, the decoupling of the organization architecture design from the data flow structure design introduces two opportunities for the fine-tuning of the C^2 organization: one at the data flow level and one at the decisionmaker and system level.

7. REFERENCES

Alaiwan, H. and J.M. Toudic (1985). Recherche des Semi-Flots, des Verrous et des Trappes dans les Reseaux de Petri. *Technique et Science Informatiques*. Vol. 4, No. 1, Dunod, France.

Andreadakis S.K. (1988). *Analysis and Synthesis of Decision-making Organizations*. Ph.D. Thesis, LIDS-TH-1740, Laboratory for Information and Decision Systems, Cambridge, Massachusetts.

Boettcher, K. L., and A. H. Levis (1982). Modeling the Interacting Decisionmaker with Bounded Rationality. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-12, No3, 334-344.

Conant, R. C. (1976). Laws of Information which Govern Systems. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-6, No 4, 240-255.

Drenick, R. F. (1986). *A Mathematical Organization Theory*. North - Holland, New York.

Fernandez C. and P.S. Thiagarajan (1984). A Lattice Theoretic View of K-Density. *Advances in Petri Nets, Lecture Notes in Computer Science*, No 84, Springer Verlag, Hamburg, FGR.

Jin, V. Y., A. H. Levis and P. Remy (1986). Delays in Acyclical Distributed Decision-making Organizations. *Proc. 4th IFAC Symposium on Large Scale Systems: Theory and Application*, Zurich, Switzerland.

Levis, A. H. (1984). Information Processing and Decision-making Organizations: A Mathematical Description. *Large Scale Systems*, Vol. 7, 151-163.

Levis, A. H., and K.L. Boettcher (1983). Modeling and Analysis of Teams of Interacting Decisionmakers with Bounded Rationality. *Automatica*, Vol. 9, No. 6, 703-709.

March, J. G. (1978). Bounded Rationality, Ambiguity, and the Engineering of Choice. *Bell Journal of Econometrics*. Vol. 9, 587-608.

Martinez, J. and M. Silva (1980). A Simple and Fast Algorithm to Obtain all Invariants of a Generalized Petri Net. In Gireaud and Reisig, (Eds), *Application and Theory of Petri Nets*. Springer Verlag, Berlin, FRG.

Miller, G.A. (1956). The magical Number Seven, Plus or Minus Two. Some Limits on our Capacity for Processing Information. *Psychological Review*, Vol. 63.

Owen, G. (1968). *Game Theory*, Saunders, Philadelphia, Pensylvania.

Peterson, J. L. (1981). *Petri Net Theory and the Modeling of Systems*. Prentice Hall, Englewood Cliffs, New Jersey.

Raisbeck, G. (1963). Information Theory: *An Introduction for Scientists and Engineers*. MIT Press, Cambridge, Massachusetts.

Ramchandani, C. (1973). *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. Technical Report No. 120. Ph.D. Thesis, Laboratory for Computer Science, MIT, Cambridge, Massachusetts.

Reisig, W. (1982). *Petri Nets: An Introduction*. Springer Verlag, New York.

Shannon, C. E., and W. Weaver (1963). *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, Illinois.

Tabak, D., and A.H. Levis (1986). Petri Net Representation of Decision Models. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-15, No 6, 812-818.